Clustering

Duen Horng (Polo) Chau
Georgia Tech

Partly based on materials by Professors Guy Lebanon, Jeffrey Heer, John Stasko, Christos Faloutsos, Le Song
Clustering in Google Image Search

How would you build this?

Video: http://youtu.be/WosBs0382SE
http://google system.blogspot.com/2011/05/google-image-search-clustering.html
Clustering in Google Search

How would you build this?
Clustering

The most common type of **unsupervised** learning

High-level idea: group **similar** things together

“**Unsupervised**” because clustering model is learned without any labeled examples
(e.g., here are some pictures of dog, group them by their breed)
Applications of Clustering

- google news
- IMDB (movie sites)
- anomaly detection
- detecting population subgroups (community detection)
  - as in healthcare
- Twitter hashtags
  - text-based clustering
- (Age detection)
Clustering techniques you’ve got to know

K-means
Hierarchical Clustering (DBSCAN)
K-means (the “simplest” technique)

Demo: [http://home.dei.polimi.it/matteucc/Clustering/tutorial_html/AppletKM.html](http://home.dei.polimi.it/matteucc/Clustering/tutorial_html/AppletKM.html)

Summary

- We tell K-means the value of $k$ (#clusters we want)
- **Randomly** initialize the k cluster “means” (“centroids”)
- **Assign** each item to the the cluster whose mean the item is closest to (so, we need a *similarity function*)
- **Update** the new “means” of all k clusters.
- If all items’ assignments do not change, stop.
K-means  What’s the catch?

Need to **decide k ourselves**.

- How to find the optimal k?

Only locally optimal (vs global)

- Different initialization gives different clusters
  
  - How to “fix” this?

- “Bad” starting points can cause algorithm to converge slowly

- Can work for relatively large dataset

- Time complexity $O(n \log n)$
Hierarchical clustering

High-level idea: build a tree (hierarchy) of clusters

Agglomerative (bottom-up)
- Start with individual items
- Then iteratively group into larger clusters

Divisive (top-down)
- Start with all items as one cluster
- Then iteratively divide into smaller clusters
Ways to calculate distances between two clusters

**Single linkage**
- minimum of distance between clusters
- similarity of two clusters = similarity of the clusters’ *most similar* members

**Complete linkage**
- maximum of distance between clusters
- similarity of two clusters = similarity of the clusters’ *most dissimilar* members

**Average linkage**
- distance between cluster centers
Hierarchical clustering for large datasets?

- OK for small datasets (e.g., <10K items)
- Time complexity between $O(n^2)$ to $O(n^3)$ where $n$ is the number of data items
- Not good for millions of items or more
- But great for understanding concept of clustering
Visualizing Clusters